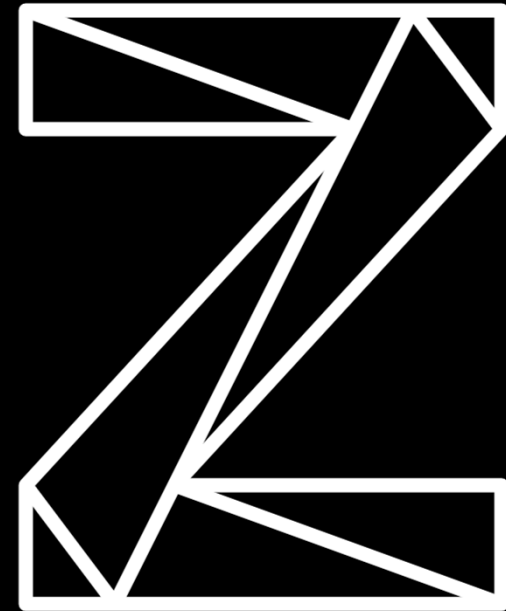


z/VM Centralized Service Management

Richard Lewis
IBM Washington System Center
rflewis@us.ibm.com



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

APPN*	GDPS*	POWERPC*	Virtual Image Facility
CICS*	Geographically Dispersed Parallel Sysplex	PR/SM	Virtualization Engine
DB2*	HiperSockets	Processor Resource/Systems Manager	VisualAge*
DFSMSMVS	HyperSwap	QMF	VM/ESA*
DFSMS/VM*	IBM*	RACF*	VSE/ESA
DirMaint	IBM eServer	Resource Link	VTAM*
Distributed Relational Database Architecture*	IBM logo*	RMF	WebSphere*
DRDA*	IBMlink	RS/6000*	z/Architecture
e-business logo*	Language Environment*	S/390*	z/OS*
ECKD	MQSeries*	S/390 Parallel Enterprise Server	z/VM*
Enterprise Storage Server*	Multiprise*	System 370	z/VSE
Enterprise Systems Architecture/390*	On demand business logo	System 390*	zSeries*
ESCON*	OS/390*	System z	zSeries Entry License Charge
FICON*	Parallel Sysplex*	System z9	
GDDM*	Performance Toolkit for VM	Tivoli*	
	POWER5	Tivoli Storage Manager	
		TotalStorage*	

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries

Linux is a trademark of Linus Torvalds in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation in the United States and other countries.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Contents

Current VM Service Process	03	Sample Scenario	17
Stand alone z/VM System	04	Initialize System for Using z/VM CSM	18
SSI Cluster	05	Identify Systems to be Managed	19
z/VM Centralized Service Management	06	Define a z/VM CSM Service Level	20
Overview and Objectives	07	Create z/VM CSM Service Level Content	21
Components and limits	08	Build z/VM CSM Service Level Package	22
z/VM CSM Detail	09	Transport Service Level Package	23
Userids and Resources	10	Test New Service Level Content	24
BASE BIN Structure and Contents	11	Transport STABLE Service Level Package	25
Service Level BIN Structure and Contents	12	Place New Service Level into Production	26
Table Definitions / SFS Directory Structure	13		
SERVMGR Command and Operands	15	Summary	27
Updates to Existing Commands, EXECS, Files	16		

Current z/VM Service Process

Standalone z/VM System

Complete service process must be executed on each standalone z/VM system.

Envelope file must be transported to each system.

SERVICE command must be run on each system.

VMSES/E maintains full inventory on each system that reflects status of each z/VM component and feature.

Put2prod must be run on each system to place new service into production.

Even though z/VM systems may be clones from a base, service must be done independently on each system.

Repeating the process numerous times is time consuming and error prone.

Systems that should be equivalent may become different due to errors in service process.

Many customers resort to copying DASD with home grown service methods.

SSI Cluster

Members of an SSI Cluster share DASD volumes.

MAINTvrn is a single configuration virtual machine that has a single context which is instantiated when logging onto any member within the cluster (only one member at a time).

The system release volume is a shared volume in an SSI cluster which contains all of the alternate service minidisks for each z/VM component and feature.

Service is installed once within the cluster on one member.

Service may be tested on any member after it has been installed using the VMSES/E SERVICE EXEC.

Service is put into production using the VMSES/E PUT2PROD command on each member when convenient.

Cluster wide utilities and files are maintained at highest service level and release level for all members.

PUT2PROD copies serviced objects from alternate disks on release volume to production runtime disks on private RES volume.

z/VM Centralized Service Management

Overview and Objectives

Extend efficiency of SSI service to non-SSI z/VM systems.

Eliminate separate receive, apply and build of service level on each non-SSI z/VM system.

Simplify roll out of system maintenance.

z/VM CSM will not affect how service is managed in SSI clusters.

Allow for handling multiple service levels.

Accommodate systems that are physically close as well as those that are distant.

Some preparation for z/VM CSM delivered in z/VM 7.1, full function delivered with z/VM 7.2.

Components and Limits

Principal system which will use SFS to manage service levels for a set of defined Managed systems.

A Principal system may manage up to 54 systems.

Principal system must not be a member of an SSI cluster.

Managed systems may not be members of an SSI cluster.

Principal system may be managed by z/VM CSM or serviced independently of z/VM CSM.

Service level will be comprised of selected IBM service deliverables such as RSU, or corrective service, or local modifications.

Service level is built upon an installed base level, or a previous service level from that base.

Shared File System is used to maintain information about managed systems, base levels, service levels.

Service levels built using the SERVICE command.

Service levels transported to Managed System using special purpose FTP servers on Managed systems.

Communication between Principal system and remote Managed systems is via FTP.

z/VM Centralized Service Management Detail

Userids and Resources

MAINTCSM – userid on Principal system to perform service management functions. Needs to be filepool administrator in VMPSFS.

CSMWORK – userid used for FTP login to a remote system for z/VM CSM transport.

MAINTvrm – release-specific service and maintenance userid, used on managed system to process z/VM CSM service management commands.

CSMSERVE – userid of dedicated FTP server for z/VM CSM. Only CSMWORK may login.

CSMvrm - release specific SFS file space ID, one for each system release level included in z/VM CSM.

VMPSFS directory structures for information pertaining to managed systems, and service level content comprising a customer defined service level.

New VMSES/E data tables:

z/VM CSM System Table – CSM SYSSTAT, containing system names, TCP/IP addresses or host names, system service level details and service status information.

z/VM CSM Service Level Table – CSM SVCSTAT, defines available service level.

Tables maintained and manipulated using VMFSIM.

BASE Structure and Contents

MAINTvrm 5BC minidisk will contain compacted file containing SFS unload of base directories.

VMPSFS file system will not initially include enough space for FILEPOOL LOAD of base directories.

Additional space (minimum of 355800 4K blocks) needs to be added to VMPSFS on the principal system before loading base directories.

Base directories will not include RSU supplied with a new system. All components will have no service applied.

FILEPOOL LOAD of CSMvrm.UNLOAD file will prime a CSMvrm file space with base directories.

Each component supplied with z/VM will have a set of directories in CSMvrm.BASE that correspond to the service SFS directories defined for components and features.

BASE ICKDSF and BASE LE directories will contain the service level that was shipped with the z/VM product.

Service Level Structure and Contents

Set of directories in a CSMvrm file space.

Top level directory for service level referenced in the z/VM CSM Service Level Table.

Service Level has directories for maintaining VMSES/E Delta, Apply and Build content for each component and feature serviced.

Service Level content consists of IBM RSU, IBM corrective (COR) service, and local modifications that might come from IBM, a vendor or customer.

VMSES/E Software Inventory files such as SRVRECS, SRVBLDS, and SRVAPPS are maintained for the Service Level and sent along with the Service Level to the managed systems.

Content from directories packaged into component-based SERVLINK files for transport to managed system.

Unique VMSESE PROFILE and CSM PPF override, maintained in top level directory for each service level to facilitate definition of test build targets.

Table Definitions / SFS Directory Structure

MAINTCSM.SYSTEM_1 is an example of the high level qualifier of a managed system. Each managed system will have an SFS directory where command log files, bitmap files, and other data pertinent to a managed system is stored.

CSMvrm.BASE is the high level qualifier for a set of directories defining the base level of z/VM components at level vrm.

Each managed system will retain its native 51D and 41D disks as well as all VMSES/E component-based SFS directories and minidisks.

Principal system will have its own distinct set of VMSES/E resources that will not be used for managing z/VM CSM service level content.

z/VM CSM related PPF variables added to VMSESE PROFILE in 7.1 to facilitate definition of z/VM CSM service levels and test build targets to build executable objects.

When new z/VM CSM service level is defined applicable :PPFVAR definitions will be updated when creating a unique VMSESE PROFILE.

CSM \$PPF file will be used to apply overrides and create a new CSM PPF file to be used instead of SERVP2P PPF for managing and building all content for a z/VM CSM service level.

Table Definitions / SFS Directory Structure (cont)

z/VM CSM service build scheme will handle building nuclei that reside in “recomped” minidisk cylinders through enhancements to the SERVICE EXEC and the new CSMPKGLD EXEC.

SERVICE command running against a z/VM CSM service level on the Principal system will use the BLDNUC user id to:

Use a temporary minidisk to hold the nucleus.

Build the nucleus to this temporary target.

DDR copy the nucleus cylinders to a file on the z/VM CSM test build SFS directory.

CSMPKGLD command running on a Managed system will:

DDR the nucleus file on the test build minidisk to the corresponding nucleus cylinders using DDR REORDER to place the cylinders in the correct “recomp” area.

PUT2PROD command running on a Managed system will:

Copy the file contents of the test build minidisk to its corresponding production minidisk.

SERVMGR Command and Operands

Command to handle all tasks associated with z/VM CSM.

SERVMGR INITIALIZE vrm – create initial structures for z/VM CSM and populate base vrm.

SERVMGR MANAGED OFF|ON – sub-function run on a managed system to apply emergency service or enable a feature.

SERVMGR SYSTEM ADD |DELETE |MODIFY |PUT2PROD |QUERY |SERVCHECK – sub-function related to maintaining a z/VM CSM system group.

SERVMGR SRVLVL ADD |DELETE |LOCALMOD |SERVICE |PACKAGE |SEND |SETLOCK | QUERY |VMFSETUP – sub-function to manage unique service level environment.

Managed system names are maintained in the z/VM CSM System Status Table.

MAINTCSM NAMES file provides for nicknames that reference multiple managed systems.

SERVMGR command activity and results stored in \$CSMCMD \$MSGLOG in the VMPSFS:CSMvrm. shared file system directory. Logs from managed systems may be stored in system specific SFS directories, for example MAINTCSM.SYSTEM_1.

Updates to Existing Commands, EXECs, Files

MAINTvrm PROFILE EXEC – includes logic to determine if logged on in disconnected state, and invoke CSMAGENT. User customization of this EXEC should now be in a separate exec that is called by PROFILE EXEC to avoid problems.

VMSESE PROFILE – add new PPFVAR variable definitions for z/VM CSM.

SERVICE EXEC – support for up to 16 character service level values, support for CSM_MANAGED tag value to prevent most SERVICE on z/VM CSM managed system.

VM SYSSUF – support new :CSM_MANAGED tag with values of YES or NO to control whether SERVICE functions can be run on a z/VM CSM managed system for a given z/VM component.

Updates to properly handle z/VM CSM named log files will be made to VMFMSG, VMFSETUP, VMFSIM, VMFVIEW, \$VMFUTL\$.

VMFTBDEF – updated to support z/VM CSM VMSES/E table files.

VMFUME REPOS updated for new messages.

Sample Scenario

Sample Scenario

Initialize System for Using z/VM CSM

Select system to serve as z/VM CSM Principal system.

Principal system is used to install and maintain all IBM-supplied PTF service as well as any local modifications to be built and supplied to z/VM CSM managed systems.

Principal system need not be managed by z/VM CSM, however, it can be if so desired.

Managed systems must be at z/VM 7.2 or higher.

```
enroll admin MAINTCSM VMPSFS:
```

Make sure VMPSFS has at least 355,800 (preliminary estimate) SFS 4K blocks available to load CSM720 UNLOAD file from MAINTCSM 5BC mdisk.

```
servmgr initialize 720
```

Creates CSM720.BASE filespace and directory structure.

Creates z/VM CSM Service Level Table and initialize with base information.

Generates service BITMAP and VVTLCL summary file for all base components.

Creates SERVLVL DESCRIPT file for BASE.

Sample Scenario

Define Initial z/VM CSM Service Level

When principal system is initialized with z/VM CSM BASE, a new service level needs to be defined (which will create a CSM service set of directories) for the initial 7.2 install RSU.

After new service level directories are created, the `servmgr srvlvl service` command can be executed to install the new RSU in the directories.

```
servmgr srvlvl add 720 vm010102_BLD1 basedon  
BASE nonesm desc Initial RSU Install
```

```
servmgr srvlvl service 720 vm010102_bld1 all  
rsu0801
```

z/VM CSM Service Level Table updated to include `vm010802_BLD1` with a TEST build state.

SFS directories created for building and storing updates comprising new service level.

New directories primed with content from the “based on” directories.

CSM PPF override created for new service level and stored in `CSM720.VM010102_BLD1`.

Sample Scenario

Identify Systems to be Managed

Managed systems cannot be members of an SSI Cluster.

Managed systems to begin with must be running z/VM 7.2.

Managed systems must be configured for z/VM CSM by configuring the CSMERVE dedicated FTP server. This is done from TCPMAINT using the CSMERVE sample CONFIG file.

Reserve ports 4534 and 4535 in PROFILE TCPIP and include CSMERVE in AUTOLOG stanza.

```
servmgr system add 720 vm01 vm010102_bld1  
nonesm hostname vm01.Washington.ibm.com  
commtype ftp (port 4535 norequire)
```

Command will confirm ftp communication, update VM SYSSUF to prevent non-z/VM CSM use of SERVICE command, and verify service level specified is correct to the system being added.

vm01 is added to the z/VM CSM System Status Table.

MAINTCSM.VM01 directory structure added to VMPSFS.

Sample Scenario

Define a new z/VM CSM Service Level

When a new RSU becomes available a new CSM service level can be created based off a previous service level or BASE.

After new service level directories are created, the `servmgr srvlvl` service command may be executed to install the new RSU in the directories.

```
servmgr srvlvl add 720 vm010802_BLD2 basedon  
vm010102_BLD1 nonesm desc August RSU Install
```

z/VM CSM Service Level Table updated to include `vm010802_BLD2` with a TEST build state.

SFS directories created for building and storing updates comprising new service level.

New directories primed with content from the “based on” directories.

CSM PPF override created for new service level and stored in `CSM720.VM010802_BLD2`.

Sample Scenario

Create z/VM CSM Service Level Content

With the new service level directories defined and populated, the new RSU content may be installed in those directories.

Obtain RSU envelope as usual from ShopZ.

If local modification processing is necessary as well for this new service level, the SRVLVL LOCALMOD command would be executed following service.

```
servmgr srvlvl service 720 vm010802_BLD2 all  
rsu0802
```

VMSES/E SERVICE Exec is used to install content of rsu0802 servlink file for all components to the directory structure created for vm010802_BLD2.

All updates and built parts are loaded.

Redirection of service processing to the z/VM CSM directories is accomplished through use of CSM PPF file generated for this particular service level.

Sample Scenario

Build z/VM CSM Service Level Package

After building the vm010802_BLD2 service level content a package can be built for transport of the content to a managed system.

Appropriate SERVLINK files will be built for each z/VM component affected by the new service level.

Additional files such as package manifest file are also included.

When the package is built, the content is considered to be in TEST build state.

```
servmgr srvlvl package 720 vm010802_BLD2
```

If it is desired to prevent any further manipulation of the content of this service level, an update control lock can be set.

```
servmgr srvlvl setlock 720 vm010802_BLD2 on
```

SERVLINK files have same format as RSU SERVLINK files although the content is significantly different

VMSES/E PARTCAT and new CSMPKG BASELIST files are compared to determine changed (base vs new) and updated files to include in SERVLINK file.

Sample Scenario

Transport Service Level Package

The `servmgr` command is used to transport the set of package files using the dedicated FTP server for z/VM CSM.

Parameters needed to establish the FTP connection are retrieved from the z/VM CSM System Status Table definition for `vm01`.

The managed system will validate the `SERVLINK` files, and then the Principal system will use an `FTP SITE` command to instruct the managed system to begin loading the `SERVLINK` content with a `CSMAGENT` command.

```
servmgr srvlvl send 720 vm010802_BLD2 vm01  
(test
```

The `CSMAGENT` command will cause the managed system to use `VMSES/E` to load the content of the `SERVLINK` files to the appropriate z/VM component SFS directories and minidisks.

On the managed system, `CSMSERVE` will `XAUTOLOG MAINT720` which starts `CSMAGENT` running in the `MAINT720` virtual machine.

`CMD` argument on `XAUTLOG` used to send command for `CSMAGENT` to execute.

Sample Scenario

Test New Service Level Content

Testing activity on the managed system is no different from that activity when service is applied and built using regular service procedures.

If necessary additional service may be incorporated into the service level being tested by unlocking the service level on the Principal system, installing the new service, repackaging the service level and sending the updated service level to the managed system.

Sample Scenario

Transport STABLE Service Level Package

After all testing is complete the service level needs to be designated as stable.

This is done by transporting the package to the managed system with an option of STABLE.

```
servmgr srvlvl send vm010802_BLD2 vm01  
(stable
```

Sending the package as stable causes the service level to be locked from further update activity, and the z/VM CSM Service Level Table is updated to reflect the status of stable.

Sample Scenario

Place New Service Level into Production

After the stable package has been processed and loaded, the service level can be put into production.

Putting the service level into production involves copying the serviced objects to the runtime production locations.

The `servmgr` command is used to invoke `put2prod` on the managed system.

```
servmgr system put2prod 720 vm01
```

The `servmgr` command establishes an FTP connection with `vm01` and uses an `FTP SITE` command to initiate `put2prod` processing on the managed system.

`CSMSERVE` on the managed system `XAUTOLOGs MAINT720`, which starts `CSMAGENT` running in `MAINT720`.

`CSMAGENT` runs `VMSES/e PUT2PROD` command after validating everything is correct.

Summary

Summary

z/VM CSM enables service to be installed once and put into production on up to 54 non-SSI managed systems.

Principal system and Managed systems may not be members of an SSI cluster.

z/VM CSM activity on principal system performed from MAINTCSM. VMPSFS file pool is used to house managed system data and service levels.

New servmgr command to perform z/VM CSM tasks.

Dedicated special purpose FTP server (CSMSERVE) used to transport data and commands from principal system to managed systems.

CSMWORK is only userid permitted to login to CSMSERVE FTP server.

Managed systems continue to use default VMSES/E directories and minidisks for service content.